MYiR 米尔电子 Make Your Idea Real

Remi Pi 实时系统与 Ethercat 移植

文件状态: [√] 草稿 [] 正式发布	**文件编号:**	MYIR-REMI PI-SW-RN-ZH-L5.10.83
	文件版本:	V1.0[文档]
	作　　者:	Mark
	创建日期:	2024-02-24
	最近更新:	2024-02-24

深圳市米尔电子有限公司 MYIR Electronics Limited
Web: www.myir.cn Mail: sales.cn@myir.cn Tel: 0755-25622735

深圳市米尔电子有限公司 MYIR Electronics Limited
Web: www.myir.cn Mail: sales.cn@myir.cn Tel: 0755-25622735

版本历史

文件版本	作者	参与者	日期	备注
V1.0[文档]	Mark	--	2024-02-24	创建该文档

深圳市米尔电子有限公司 MYIR Electronics Limited
Web: www.myir.cn Mail: sales.cn@myir.cn Tel: 0755-25622735

目 录

1. 概述

Remi Pi 采用瑞萨 RZ/G2L 作为核心处理器，该处理器搭载双核 Cortex-A55@1.2GHz+Cortex-M33@200MHz 处理器，其内部集成高性能 3D 加速引擎 Mail-G31 GPU(500MHz)和视频处理单元（支持 H.264 硬件编解码）,16 位的 DDR4-1600 / DDR3L-1333 内存控制器、千兆以太网控制器、USB、CAN、SD 卡、MIPI-CSI 等外设接口，在工业、医疗、电力等行业都得到广泛的应用。

- 在开发阶段，建议配合核心板配套的评估套件 MYD-YG2L23-8E1D-120-C-REMI 来加速开发。评估套件的详细信息请访问：
  https://www.myir.cn/shows/23/14.html

深圳市米尔电子有限公司 MYIR Electronics Limited
Web: www.myir.cn Mail: sales.cn@myir.cn Tel: 0755-25622735

2. 实时内核设计

●实时补丁我们选择 RT-Preempt 来实现

2.1. 移植补丁

RT 补丁可以从 RT 官网下载 5.10.83 对应的补丁
https://cdn.kernel.org/pub/linux/kernel/projects/rt/5.10/older/

●把补丁放到自己的工作目录下，然后解压，如图 2-1：

```
hjx@myir-server:~/renesas/04_Sources$ tar -xvf patches-5.10.83-rt58.tar.gz
```

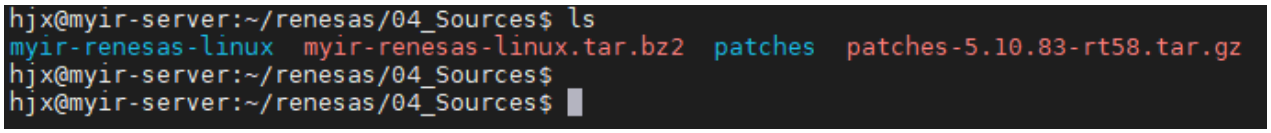

图 2-1. 内核源码解压

●进入到内核源码打补丁，如图 2-2：

```
hjx@myir-server:~/renesas/04_Sources/myir-renesas-linux$ for p in `ls -1 ../patches/*.patch`; do patch -p1 < $p; done
```

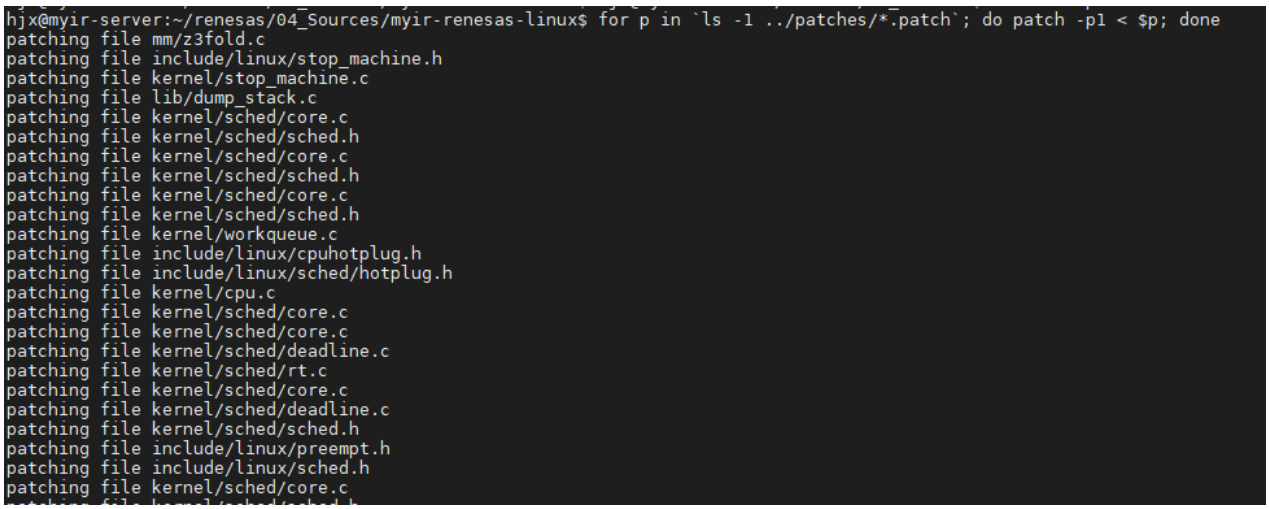

图 2-2. 打实时补丁

●编译内核源码

```
hjx@myir-server:~/renesas/04_Sources/myir-renesas-linux$ /opt/remi-sdk/environment-setup-aarch64-poky-linux
```

```
hjx@myir-server:~/renesas/04_Sources/myir-renesas-linux$ make ARCH=arm64
mys_g2lx_defconfig
hjx@myir-server:~/renesas/04_Sources/myir-renesas-linux$ make ARCH=arm64 I
mage dtbs   -j16
```

更新编译得到的 Image 即可

2.2. 其他影响性能的配置

禁用 CPU Freq 自动调频，并设置主频为最高频率：

```
cd /sys/devices/system/cpu/cpufreq/policy0
echo userspace > scaling_governor
cat scaling_max_freq  > scaling_setspeed
```

（如不禁用 cpufreq 调频功能，系统会因动态调频产生极大的偶然延迟）

3. 实时性测试

●空载测试

```
cyclictest -p 99 -t 1 -d 100 -i 1000 -D 24h -m -a -n
```

```
[root@myir:/]# cyclictest -p 99 -t 1 -d 100 -i 1000 -D 24h -m -a -n
# /dev/cpu_dma_latency set to 0us
policy: fifo: loadavg: 0.11 0.03 0.01 1/155 1478
policy: fifo: loadavg: 0.06 0.03 0.00 1/155 1478
policy: fifo: loadavg: 0.03 0.02 0.00 1/155 1478          vg:    8[   44.843983] nf_con
T: 0 ( 1477) P:99 I:1000 C:  41816 Min:    72.148095] random: crng init done  24
T: 0 ( 1477) P:99 I:1000 C:  82684 Min:      6 Act:    8 Avg:    8 Max:      24
```

图 3-1.空载测试

●CPU&内存满载

```
cyclictest -p 99 -t 1 -d 100 -i 1000 -D 24h -m -a -n
```

●增加压力

```
stress-ng --cpu 4 --cpu-method all --io 4 --vm 50 -d 5  --fork 4 --timeout 36000s
```

```
Mem: 997944K used, 12892K free, 2592K shrd, 1724K buff, 665196K cached
CPU:  87% usr  12% sys   0% nic   0% idle   0% io   0% irq   0% sirq
Load average: 43.26 12.45 4.28 61/276 5826
  PID  PPID USER     STAT   VSZ %VSZ %CPU COMMAND
 1551  1509 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1592  1570 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1525  1516 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1526  1497 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1582  1560 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1532  1500 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1583  1563 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1523  1514 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1550  1537 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1545  1507 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1618  1576 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1620  1607 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1484  1481 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1541  1502 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1584  1558 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1601  1589 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1586  1569 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1587  1565 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1610  1594 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
 1531  1513 root     R    18380   2%   2% {stress-ng-vm} stress-ng --cpu 4 --cpu
[root@myir:/]# cyclictest -p 99 -t 1 -d 100 -i 1000 -D 24h -m -a -n
# /dev/cpu_dma_latency set to 0us
policy: fifo: loadavg: 68.39 56.73 29.71 59/273 12339 [  711.336471] sunxi-gmac gmac1 eth1: Link isn
policy: fifo: loadavg: 68.39 56.73 29.71 60/275 12440
policy: fifo: loadavg: 68.28 65.45 43.76 60/272 13755           13 Max:     100
policy: fifo: loadavg: 68.28 65.45 43.76 60/272 13862          ac1 eth1: Link is Up - 100Mbps/Full f
policy: fifo: loadavg: 68.27 66.82 48.51 62/274 30282          Link is Down 136
policy: fifo: loadavg: 68.33 66.86 48.62 65/275 30390           13 Max:  [ 1134.250982] sunxi-gmac f
policy: fifo: loadavg: 68.42 68.32 64.22 62/273 18324          Link is Down 136
T: 0 ( 6434) P:99 I:1000 C:1084953 Min:      6 Act:   13 Avg:   13 Max:  [ 1335.979872] sunxi-gmac f
T: 0 ( 6434) P:99 I:1000 C:2540772 Min:      6 Act:   11 Avg:   13 Max:     136
```

图 3-2.满载测试

数据对比：

板卡	MYD-YG2LX-REMI

深圳市米尔电子有限公司 MYIR Electronics Limited
Web: www.myir.cn Mail: sales.cn@myir.cn Tel: 0755-25622735

测试时间	120min
指令	cyclictest &stress-ng
空载	平均 8us 最大 24us
满载	平均 13us 最大 136us

表 4-1.数据信息

4. ethercat IGH 移植

4.1. 下载 ethercat IGH 源码

到官网下载 1.5 版本的 ethercat 源码，如图 4-1：
https://gitlab.com/etherlab.org/ethercat/-/tree/stable-1.5?ref_type=heads

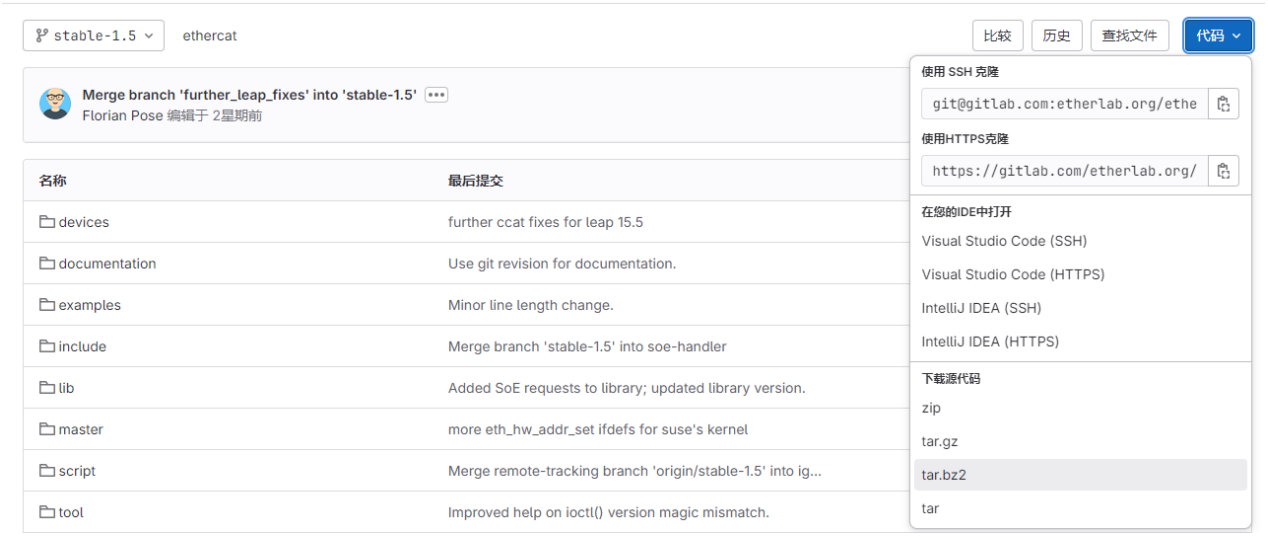

图 4-1. 下载源码

●解压 ethercat 源码

```
hjx@myir-server:~/renesas/04_Sources$ tar -xvf ethercat-stable-1.5.tar.bz2
hjx@myir-server:~/renesas/04_Sources$ cd ethercat-stable-1.5
```

●加载 sdk 环境变量

```
hjx@myir-server:~/renesas/04_Sources/myir-renesas-linux$ source /opt/remi-sdk
/environment-setup-aarch64-poky-linux
```

4.2. 编译 ethercat 源码

●生成 configure 文件

```
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$ ./bootstrap
+ touch ChangeLog
+ mkdir -p m4
+ autoreconf -i
libtoolize: putting auxiliary files in AC_CONFIG_AUX_DIR, 'autoconf'.
libtoolize: copying file 'autoconf/ltmain.sh'
libtoolize: putting macros in AC_CONFIG_MACRO_DIRS, 'm4'.
```

```
libtoolize: copying file 'm4/libtool.m4'
libtoolize: copying file 'm4/ltoptions.m4'
libtoolize: copying file 'm4/ltsugar.m4'
libtoolize: copying file 'm4/ltversion.m4'
libtoolize: copying file 'm4/lt~obsolete.m4'
configure.ac:56: installing 'autoconf/ar-lib'
configure.ac:55: installing 'autoconf/compile'
configure.ac:58: installing 'autoconf/config.guess'
configure.ac:58: installing 'autoconf/config.sub'
configure.ac:42: installing 'autoconf/install-sh'
configure.ac:42: installing 'autoconf/missing'
examples/dc_user/Makefile.am: installing 'autoconf/depcomp'
```

●configure 设置

```
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$ mkdir  output
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$ ./configure --prefix=/
home/hjx/renesas/04_Sources/ethercat-stable-1.5/output --with-linux-dir=/home
/hjx/renesas/04_Sources/myir-renesas-linux  --enable-8139too=no  --enable-gen
eric=yes   --host=aarch64-poky-linux
```

（--prefix=/home/hjx/renesas/04_Sources/output 指定输出目录、--with-linux-dir=/home/hjx/renesas/04_Sources/myir-renesas-linux 指定内核源码目录）

```
configure: loading site script /opt/remi-sdk/site-config-aarch64-poky-linux
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for aarch64-poky-linux-strip... aarch64-poky-linux-strip
checking for a thread-safe mkdir -p... /bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether make supports nested variables... yes
checking whether make supports nested variables... (cached) yes
checking for a sed that does not truncate output... (cached) sed
checking for aarch64-poky-linux-pkg-config... no
checking for pkg-config... /opt/remi-sdk/sysroots/x86_64-pokysdk-linux/usr/bin/p
kg-config
```

```
checking pkg-config is at least version 0.9.0... yes
checking whether make supports the include directive... yes (GNU style)
.....
.............
config.status: creating examples/xenomai/Makefile
config.status: creating examples/xenomai_posix/Makefile
config.status: creating include/Makefile
config.status: creating lib/Makefile
config.status: creating lib/libethercat.pc
config.status: creating master/Kbuild
config.status: creating master/Makefile
config.status: creating script/Makefile
config.status: creating script/init.d/Makefile
config.status: creating script/init.d/ethercat
config.status: creating script/sysconfig/Makefile
config.status: creating tool/Makefile
config.status: creating tty/Kbuild
config.status: creating tty/Makefile
config.status: creating config.h
config.status: executing depfiles commands
config.status: executing libtool commands
```

●编译源码

```
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$   make

make  all-recursive
make[1]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5'
Making all in include
make[2]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/in
clude'
make[2]: Nothing to be done for 'all'.
make[2]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/inc
lude'
Making all in script
```

```
make[2]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/sc
ript'
Making all in init.d
make[3]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/sc
ript/init.d'
make[3]: Nothing to be done for 'all'.
make[3]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/scr
ipt/init.d'
Making all in sysconfig
make[3]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/sc
ript/sysconfig'
make[3]: Nothing to be done for 'all'.
make[3]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/scr
ipt/sysconfig'
......
......
make[3]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/ex
amples/user'
make[3]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/ex
amples'
make[3]: Nothing to be done for 'all-am'.
make[3]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/ex
amples'
make[2]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/ex
amples'
make[2]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5'
make[2]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5'
make[1]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5'
```

●编译 modules

```
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$  make  modules

make[1]: Entering directory '/home/hjx/renesas/04_Sources/myir-renesas-linux'
```


深圳市米尔电子有限公司 MYIR Electronics Limited
Web: www.myir.cn Mail: sales.cn@myir.cn Tel: 0755-25622735

```
  CC [M]  /home/hjx/renesas/04_Sources/ethercat-stable-1.5/examples/mini/mini.
o
  LD [M]  /home/hjx/renesas/04_Sources/ethercat-stable-1.5/examples/mini/ec_mi
ni.o
  CC [M]  /home/hjx/renesas/04_Sources/ethercat-stable-1.5/master/cdev.o
  CC [M]  /home/hjx/renesas/04_Sources/ethercat-stable-1.5/master/coe_emerg_ri
ng.o
  CC [M]  /home/hjx/renesas/04_Sources/ethercat-stable-1.5/master/datagram.o
.....
.....
make[1]: Leaving directory '/home/hjx/renesas/04_Sources/myir-renesas-linux'
```

编译通过会对应生成 ethercat-stable-1.5/devices/ec_generic.ko 和 ethercat-stable-1.5/master/ec_master.ko

4.3. 安装 ethercat

安装成功后前面指定/home/hjx/renesas/04_Sources/ethercat-stable-1.5/output 目录下有编译生成的各种用户空间的文件。

```
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$ make install
Making install in include
make[1]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/in
clude'
make[2]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/in
clude'
make[2]: Nothing to be done for 'install-exec-am'.
 /bin/mkdir -p '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/output/include
'
 /usr/bin/install -c -m 644 ecrt.h ectty.h '/home/hjx/renesas/04_Sources/ethercat-s
table-1.5/output/include'
make[2]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/inc
lude'
make[1]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/inc
lude'
Making install in script
```

```
make[1]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/script'
Making install in init.d
make[2]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/script/init.d'
make[3]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/script/init.d'
make[3]: Nothing to be done for 'install-exec-am'.
.....
.....
make[3]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/examples'
make[2]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/examples'
make[1]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5/examples'
make[1]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5'
make[2]: Entering directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5'
make[2]: Nothing to be done for 'install-exec-am'.
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5'
make[1]: Leaving directory '/home/hjx/renesas/04_Sources/ethercat-stable-1.5'
```

4.4. 将 etharcat 相关文件打包

在/home/hjx/renesas/04_Sources/ethercat-stable-1.5/output 目录下创建 modules 文件夹，并复制 ec_generic.ko 和 ec_master.ko 到 modules 下

```
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$  mkdir -p output/modules
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$  cp devices/ec_generic.ko  output/modules/
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$  cp master/ec_master.ko   output/modules/
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$ ls output
```

```
bin  etc  include  lib  modules  sbin share
```

●压缩 output 输出文件

```
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$ tar -jcvf output.tar.bz
2 output/
output/
output/etc/
output/etc/ethercat.conf
output/etc/init.d/
output/etc/init.d/ethercat
output/etc/sysconfig/
output/etc/sysconfig/ethercat
output/sbin/
output/sbin/ethercatctl
output/include/
output/include/ectty.h
output/include/ecrt.h
output/modules/
output/modules/ec_master.ko
output/modules/ec_generic.ko
output/share/
output/share/bash-completion/
output/share/bash-completion/completions/
output/share/bash-completion/completions/ethercat
output/bin/
output/bin/ethercat
output/lib/
output/lib/libethercat.so
output/lib/pkgconfig/
output/lib/pkgconfig/libethercat.pc
output/lib/libethercat.so.1.2.0
output/lib/cmake/
output/lib/cmake/ethercat/
output/lib/cmake/ethercat/ethercat-config.cmake
```

```
output/lib/libethercat.a
output/lib/libethercat.la
output/lib/libethercat.so.1
hjx@myir-server:~/renesas/04_Sources/ethercat-stable-1.5$ ls output.tar.bz2
output.tar.bz2
```

至此 IGH 交叉编译完成，下面是在对应 arm 目标板上的操作。

4.5. 移植 ethercat 相关库到开发板

将上面制作好的 output.tar.bz2 传到开发板上，然后解压出来。

```
root@myir-remi-1g:~# tar -xvf output.tar.bz2
output/
output/etc/
output/etc/ethercat.conf
output/etc/init.d/
output/etc/init.d/ethercat
output/etc/sysconfig/
output/etc/sysconfig/ethercat
output/sbin/
output/sbin/ethercatctl
output/include/
output/include/ectty.h
output/include/ecrt.h
output/modules/
output/modules/ec_master.ko
output/modules/ec_generic.ko
output/share/
output/share/bash-completion/
output/share/bash-completion/completions/
output/share/bash-completion/completions/ethercat
output/bin/
output/bin/ethercat
output/lib/
output/lib/libethercat.so
```

```
output/lib/pkgconfig/
output/lib/pkgconfig/libethercat.pc
output/lib/libethercat.so.1.2.0
output/lib/cmake/
output/lib/cmake/ethercat/
output/lib/cmake/ethercat/ethercat-config.cmake
output/lib/libethercat.a
output/lib/libethercat.la
output/lib/libethercat.so.1
```

将 output 目录下各文件目录的内容复制到板子根文件系统根目录下相应目录下，例如：cp bin/ethercat /bin/(include 目录不用复制)

```
root@myir-remi-1g:~/output# ls
bin  etc  include  lib  modules  sbin  share
root@myir-remi-1g:~/output# cp bin/ethercat /bin/
root@myir-remi-1g:~/output# cp etc/ethercat.conf /etc/
root@myir-remi-1g:~/output# cp etc/init.d/*  /etc/init.d
root@myir-remi-1g:~/output# cp -r etc/sysconfig/ /etc/
root@myir-remi-1g:~/output# cp  lib/libethercat.* /lib64/
root@myir-remi-1g:~/output# cp -r lib/pkgconfig /lib64/
root@myir-remi-1g:~/output# cp modules/ec_master.ko /lib/modules/5.10.83-cip
1-yocto-standard/
root@myir-remi-1g:~/output# cp sbin/ethercatctl  /sbin/
```

4.6. 启动 EtherCAT

4.6.1. 配置主站的 MAC 地址

```
root@myir-remi-1g:~# depmod
root@myir-remi-1g:~# modprobe ec_master main_devices=1E:ED:19:27:1A:B3
```

4.6.2. 启动 ethercat

```
root@myir-remi-1g:~# /etc/init.d/ethercat start
Starting EtherCAT master 1.5.2  done
```

至此所有步骤完成。